COLLEGE OF ENGINEERING
UNIVERSITY of WASHINGTON

COMPUTER VISION

# Intracranial Artery Segmentation Using Convolutional Autoencoder

Li Chen, Yanjun Xie

## Background

Finding an robust and effective way to segment and trace intracranial artery features from Time-of-flight (TOF) magnetic resonance angiography (MRA) is essential for

- Study cerebral blood supply
- Finding intracranial artery variations
- Evaluation of intracranial vascular features

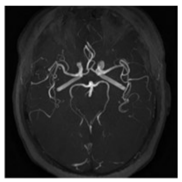
Accurate vessel segmentation method is the critical step.

## Task

Finding intracranial arteries with convolutional neural network.

## Challenges

- 3D image segmentation
- Limited number (49) of MRA images
- Hard to label ground truth and may not be perfect
- Many parameters to adjust in neural network



Original TOF MRA image from Maximum Intensity Projection view

Label image as ground truth

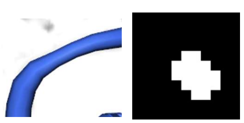
## METHOD

Dataset:

- 49 sets of 3D MRA images (size: 620*620*243, same imaging parameters from same machine)
- Arteries labeled previously (traces of center points and radii)
- Training set: 46 images
- Validation set: 2 images
- Testing set: 1 image

Preprocess

Generate Label image: 3D traces rendered in VTK with Laplacian smoothing and voxelized to label image, with 1 for artery, 0 for non-artery.



Left: Artery rendered in VTK
Right: One slice of label image patch

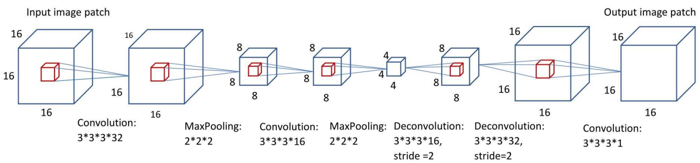
Network structure:

8-layer Convolutional autoencoder (CAE)

- Encoder: represent extracted features with latent variables
- Decoder: produce a result with latent variables.

Contact

| | |
|---|---|
| Li Chen | Email: cluw@uw.edu |
| Yanjun Xie | Email: yanjunx@uw.edu |


Network Structure of Convolutional Autoencoder

Training

Use patches of 16*16*16 extracted from 3D original image and label image as input and output:

- Regional similarities of artery structure
- Limited images available, much more if cutting into patches
- Memory and calculation concern, over 93M voxels each image
- Largest artery diameter rarely exceeds 16 pixels

Extracting patches with different sliding strides to balance samples between artery and non-artery patches

Loss function: binary cross-entropy        Optimizer: Adam

Training using ~15k patches from one image at a time, trained 5 epochs before next image. Repeat when all images trained.

Predicting

Predict from patches with overlap for smoothing
Thresholding for binary classification
Morphometry operation: Close (dilate+erode) connect broken area

Parameter optimization

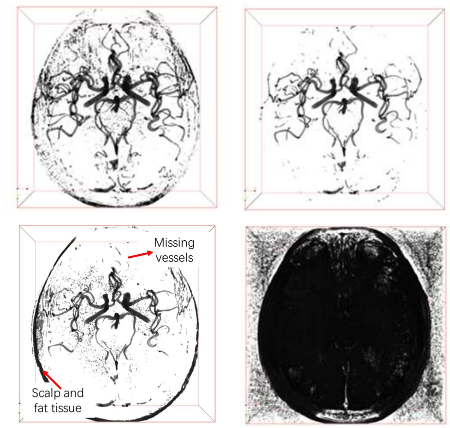
Tune a list of parameters to improve performance (see Table 1)
Evaluated and chosen considering

- Top Youden index (True Positive Rate – False Positive Rate)
- Area Under Curve (AUC) from ROC curve
- Highest accuracy with best threshold
- Calculation and memory cost for implementation

## RESULT

Table 1: Performance between different parameters for training neural network, final choice marked in bold

| Parameter type | Parameter selection | Youden index | AUC | Highest accuracy |
|---|---|---|---|---|
| Activation function | **ReLU** | 0.91459 | 0.98916 | 0.99707 |
| | Tanh | 0.90775 | 0.98753 | 0.99699 |
| | Sigmoid | 0.00210 | 0.49904 | 0.99504 |
| Convolution layers | 1 | 0.90813 | 0.98957 | 0.99709 |
| | **2** | 0.91767 | 0.99267 | 0.99731 |
| | 3 | 0.91698 | 0.99211 | 0.99715 |
| Kernel numbers | 16, 8 | 0.91836 | 0.99221 | 0.99718 |
| | **32,16** | 0.91767 | 0.99267 | 0.99731 |
| | 64, 32 | 0.90115 | 0.98874 | 0.99734 |
| Up-sample method | Up-pooling | 0.91375 | 0.99272 | 0.99731 |
| | **Deconvolve with stride** | 0.91446 | 0.99299 | 0.99716 |
| Down-sample method | **Max-pooling** | 0.91698 | 0.99211 | 0.99715 |
| | Convolve with stride | 0.89896 | 0.98941 | 0.99692 |
| Images trained | 15 | 0.87975 | 0.98995 | 0.99682 |
| | **15+31** | 0.91297 | 0.99302 | 0.99740 |
| Repetition | 1 | 0.90890 | 0.99067 | 0.99713 |
| | **2** | 0.91865 | 0.99244 | 0.99730 |




3D rendered segmentation results of
Top left: Probability image predicted from CAE
Top right: Binary image after thresholding predicted image
Bottom left: Renyi Entropy [1] thresholded image
Bottom right: Phansalkar [2] thresholded image

Train with optimized parameter on work station (Intel® Xeon® CPU E5-2630 v3 @2.4GHz 8 cores, 32 GB Memory, NVIDIA GeForce GTX TITAN X on Windows 7)
About 1 minute per image per epoch, 592 minutes in total
Compare with best threshold and local threshold method provided in ImageJ, binary classification performance shown in Table 2

Table 2: Comparison of binary classification performance with traditional methods

| | Accuracy | Sensitivity | Specificity | Precision |
|---|---|---|---|---|
| CAE | 0.997413 | 0.578091 | 0.999503 | 0.852728 |
| CAE with close | 0.997419 | 0.580661 | 0.999495 | 0.851610 |
| Renyi Entropy [1] | 0.994823 | 0.504220 | 0.997268 | 0.479052 |
| Phansalkar [2] | 0.866895 | 0.932149 | 0.866570 | 0.033638 |

## Conclusions

The structure of 8-layer autoencoder with optimized parameters was effective in segmentation of intracranial artery from TOF MRA images with binary classification performance better than two traditional image processing methods

## Future work

Combining patch origin into neural network
Training with more data
Test with other machine learning algorithms: denoised autoencoder, adversarial autoencoder, variationally autoencoder.

## Reference

[1] J. N. Kapur, Comput. Vision, Graph. Image Process, 1985.
[2] N. Phansalkar, ICCSP 2011.